

AMENDMENT TO THE SPECIFICATION:

Paragraph starting at line 16 of page 1 has been amended as follows:

The aforementioned user program is usually created by a ladder program capable of using function blocks established according to JIS B3503 (1997). Input and output parameters, internal variables and operations (~~algorithms~~) (algorithms) of the function block are established by using a language element called function block definition. When a function block is incorporated in a user program, copies referred to as function block instances are created by instantiating (or invoking) this function block definition and such function block instances are written in a ladder program. Thus, when instances based on the same function block definition are used at a plurality of locations, each is instantiated such that a plurality of function block instances are created and incorporated at each of the localities within the program. Each function instance is given its unique instance name and identified uniquely on the basis of its instance name. A function block definition may be defined so as to call another function block definition in its ~~algorithm~~ algorithm.

Paragraph starting at line 31 of page 1 has been amended as follows:

Function block definitions and functional block instances will be explained next by way of an example. Let us assume that function block definitions A and B are defined as shown in Fig. 1 which is a ladder diagram showing the content of the definitions of the ~~algorithm~~ algorithm of the function block definition. As can be seen from this definition, function block definition A is calling different function block definition B. When these function block definitions A are actually used in a user program, they are instantiated with into function block instances with instance names I-1 and I-2. Function block definitions B are also invoked although their instance names are not shown. Fig. 1 shows an example where two instances (copies) with instance names I-1 and I-2 are created from one function block definition A and one function block instance B is called from one function block instance A.

Paragraph starting at line 1 of page 3 has been amended as follows:

Programming devices (or display and edit devices) for creating new programs and editing (or debugging) already created programs are adapted to display a program as it is examined and edited. Fig. 4 shows an example of a displayed function block, a list of function block definitions appearing on the left-hand side and a detailed definition of a selected function block definition on the right-hand side. In the example of Fig. 4, the content of definitions in the ~~algorism~~ algorithm of the function block definition is displayed as a ladder diagram.

Paragraph starting at line 24 of page 3 has been amended as follows:

Secondly, although the ~~algorism~~ algorithm (the ladder diagram in the example of Fig. 4) displayed on the right-hand side of the display screen could be either a function block definition of a function block instance, the displayed ladder diagram was the same for both a function block definition and a function block instance. Thus, it was not easy to understand by a quick glance at the display whether what is being displayed is a function block definition or a function block instance.

Paragraph starting at line 11 of page 4 has been amended as follows:

Fifthly, it was difficult while the ~~algorism~~ algorithm of a function block instance is being displayed on the display screen to call the declaration of the function block definition from which the copy was made, say, for the purpose of editing.

Paragraph starting at line 16 of page 4 has been amended as follows:

It is therefore an object of this invention to provide a display and edit device, a

display method and a program product with improved workability, making it possible to easily understand the structure relationships of function block definitions and function block instances, whether the ~~algorism~~ algorithm, etc. on the display screen are displaying a function block definition of function block instance, and which of a plurality of function block instances based on a single function block definition is being displayed, and being capable of making a display (such as screen for editing a function block definition or for monitoring a function block instance) suitable for the current process and calling a related function block definition from a displayed function block definition.

Paragraph starting at line 26 of page 5 has been amended as follows:

It is preferable to further provide the display and edit device of this invention with a display judging device for causing the structure display device to

display with an emphasis the structure relationship of the selected function block definition or selected function block instance or a corresponding portion of the structure relationship of the function block instance. With the device thus provided, the user can easily understand whether contents of definitions such as an ~~algorism~~ algorithm by a function block displayed on the display screen are based on a function block definition or a function block instance and, if there are a plurality of function block instances based on a same function block definition, which of the instances is being displayed.

Paragraph starting at line 15 of page 7 has been amended as follows:

The instance analyzer 15 is for detecting a structure relationship of function block instances as shown in Fig. 3. Explained more in detail, a program temporarily stored in the program memory 12 is searched from its start and, when a function block instance is detected, its instance name is obtained and the ~~algorism~~ algorithm of this function block instance is analyzed to judge the presence or absence of any function block instance being called. If there is a function block instance that is being called, this function block instance being called is connected below the original function block instance. Distinctions among

function block instances are made by way of their instance names that are uniquely assigned.

Paragraph starting at line 3 of page 8 has been amended as follows:

The definition analyzer 16 is for detecting a structure relationship of function block definitions as shown in Fig. 2. Explained more in detail, a program temporarily stored in the program memory 12 is searched from its start and, when a function block definition is detected, its definition name is obtained and the internal variables (internal circuits) of this function block definition are analyzed to judge the presence or absence of any function block definition being called. If there is a function block definition that is being called in the ~~algorism~~ algorithm, this function block definition being called is connected below the original function block definition. If there is a function block definition that is being called, this function block definition being called is further analyzed to determine whether still another function block definition is being called thereby. This is continued, and when a function block definition that is not making any call is reached, a connection relationship can be established among this series of function block definitions.

Paragraph starting at line 25 of page 8 has been amended as follows:

Explained more in detail, the left-hand side of the display screen serves to display the structure relationship of function blocks and the left-hand side of the screen serves to display the ~~algorism~~ algorithm (a ladder diagram in the examples of Figs. 6 and 7) of the selected function block for monitoring and editing. Not only the structure relationship of function block definitions as before but also the tree-form of the structure relationship of function block instances is displayed in stages on the left-hand side. The display of the structure relationship may alternatively be made in the form shown in Fig. 2. If any of the function block definitions or function block instances displayed on the left-hand side is clicked upon through an input device (or a pointing device) 21, the ~~algorism~~ algorithm of the selected function block definition or function block instance is displayed on the right-hand side.

Paragraph starting at line 11 of page 9 has been amended as follows:

If a function block instance is selected, a block instance display device 20 detects the input and responds by reading out the corresponding function block instance from the program memory 12, obtaining I/O data from the data memory 13 and displaying the operating conditions of the ~~algorism~~ algorithm of the function block instance based on the obtained I/O data as shown in Fig. 7. Since Fig. 7 shows the ~~algorism~~ algorithm of the function block instance by a ladder diagram, its operating conditions are displayed by emphasizing a portion of such a ladder diagram. In this situation, a display judging device 18 ascertains which of the function block definitions or function block instances in the displayed structure relationship has been selected and transmits a command to the block structure display device 17 to display the selected portion with an emphasis. Upon receiving this command, the block structure display device 17 makes the requested display with an emphasis. Thus, the user has only to glance over the emphasized portion on the left-hand side of the display screen in order to ascertain whether it is a function block definition or a function block instance that is currently being displayed on the right-hand side of the display screen and, if it is a function block instance, which of the function block instances.